**03 – FLUSSO APPLICATIVO AD ALTO LIVELLO IN VULKAN**

* **Inizializzazione Engine**

Per inizializzare Vulkan, crei una **VkInstance** dove recuperare la lista dei **VkPhysicalDevice**, ovvero tutte le GPU disponibili nel computer. Per ogni dispositivo viene creato un oggetto di questo tipo.

Dopo averne recuperato i dettagli, crei un **VkDevice** da essi, per poi ottenerne il **VkQueue** per eseguire i comandi.

In seguito inizializzi una **VkSwapchainKHR.**

Insieme alle **VkQueue** crei delle **VkCommandPool** che ti permettono di allocare i command buffer.

* **Inizializzazione Asset**

Una volta inizializzate le strutture, inizializzi le risorse necessarie per renderizzare qualunque cosa. I materiali sono caricati e crei un set di **VkPipeline** per le combinazioni e parametri di shader necessari a renderizzare i materiali.

Per le **mesh** (modelli 3D), carichi i dati dei loro vertici nei **VkBuffer** e le loro texture in **VkImage,** assicurandoti che l’immagine sia in una disposizione leggibile

* **Render Loop**

Ora che è tutto pronto per il rendering, chiedi a **VkSwapchainKHR** un’immagine da renderizzare, poi allochi memoria ad un **VkCommandBuffer** tramite una **VkCommandBufferPool** associata, poi creiamo un loop di rendering e inoltriamo il command buffer alla queue per renderizzare sulla GPU.

Infine, se il command buffer ha terminato l’esecuzione lo rimuoviamo.